AMENDMENTS TO THE SPECIFICATION

Replace paragraph [0006] with the following paragraph:

[0006] Some systems integrate personal computing with a TV to display content. <u>Interactive TV</u> <u>and Television-based internet access products, such as WEB TV™,</u> ~~Products such as WEB~~ ~~WebTV™~~ integrate web browsing and e-mail features with a TV. In other systems, a personal computer (PC) is connected to an Internet service provider (ISP) that provides the content for the web browsing and e-mail features. Software programs, such as the e-mail program, tend to be small and easily stored. Those skilled in the art recognize that these PCs do not provide adequate security such that they are susceptible to viruses and hackers.

Replace paragraph [0032] with the following paragraph:

[0032] The present invention uses ciphertext tokens to enhance authorization in a television (TV) set top box. An object is stored in two portions where one portion is inaccessible before authorization. <u>Encryption</u> ~~Decryption~~ can be used to make the one portion inaccessible. The set top box decrypts the one portion after authorization to reformulate and use the object.

Replace paragraph [0065] with the following paragraph:

[0065] The software object **608** includes content the system **200** is designed to deliver to set top boxes **108**. Several types of information can be embedded in a software object, such as executable programs, firmware upgrades, run-time programs (e.g., ~~Java®~~ <u>JAVA®</u> or <u>ACTIVE</u> <u>X®</u> ~~ActiveX®~~), programming schedules, billing information, video, audio, or data. The software object **608** can be used immediately after authentication and authorization or at a later time. Additionally, authorization can be programmed to expire after a certain amount of time.

Replace paragraph [0115] with the following paragraph:

[0115] The above discussion relates to running applications or objects **608** on an OS. These concepts are equally applicable to <u>run-time programs such as</u> ~~Java®~~ JAVA® applications running on a ~~Java®~~ JAVA® virtual machine (JVM) which runs on top of the OS. To aid in this abstraction, the concept of superordination and subordination are explained in relation to FIG. 17. Superordination and subordination define which object **608** has the responsibility to impose a checkpoint upon another object. Checkpoints are imposed on objects **608** during the normal interaction that occurs with other objects **608** and resources.